

REGULACIÓN DE UN INNOSAT MEDIANTE ALGORITMOS EVOLUTIVOS

Juan Fernando García Mejía¹, José Antonio García Mejía¹, Allan Antonio Flores Fuentes, Rafael Valentín Mendoza Mendez², Gisela Regina Baena Castro², Ricardo Vázquez Robledo³

¹Universidad Autónoma del Estado de México

Centro Universitario UAEM Atlacomulco

Email: fgarciam@uaemex.mx, jagarciame@uaemex.mx, aafloresf@uaemex

²Universidad Autónoma del Estado de México

Centro Universitario UAEM Temascaltepec

Email: rvmendozam@uaemex.mx, grbaenac@uaemex.mx

³Agencia Espacial Mexicana

Centro Regional de Desarrollo Espacial Atlacomulco

Email: vazquez.ricardo@aem.gob.mx

RESUMEN

El InnoSAT, es un nanosatélite desarrollado por la Agencia Espacial de Malasia (ANGKASA) para realizar actividades de difusión, investigación y desarrollo tecnológico, como el diseño de reguladores que controlan su comportamiento en el espacio, especialmente en los ejes de oscilación y giro, que según sus funciones de transferencia tienen un comportamiento marginalmente estable y afectan su altitud. una situación que requiere una corrección a través del diseño e implementación de técnicas de control automático como los reguladores lineales cuadrados (LQR), que se diseñan a través de la optimización de una función de costo utilizando una matriz de ponderación. Aunque esta técnica garantiza la estabilidad del sistema, a menudo se asocia con un error cuadrático medio alto y un tiempo de configuración lento. Esta propuesta presenta una alternativa para mejorar el diseño de reguladores a través de técnicas evolutivas como el Algoritmo Genet-ic y la Búsqueda de Cuckoo. Estos enfoques reducen los errores y aceleran el tiempo de configuración en los ejes oscilantes y oscilantes InnoSAT.

Palabras Clave: Regulación de altitud, Búsqueda de Cuckoo, Algoritmos genéticos, Nanosatélites

ABSTRACT

The InnoSAT, is a nanosatellite developed by the Malaysian Space Agency (ANGKASA) to carry out dissemination, research and technological development activities, such as the design of regulators that control its behavior in space, especially in the swing and spin axes, which according to their transfer functions have a marginally stable behavior and affect their altitude. a situation that requires a correction through the design and implementation of automatic control techniques such as square linear regulators (LQRs), which are designed through the optimization of a cost function using a weighting matrix. Although this technique ensures system stability, it is often associated with high mean square error and slow setup time. This proposal presents an alternative to improve the design of regulators through evolutionary techniques such as the Genetic Algorithm and the Cuckoo Search. These approaches reduce errors and speed up setup time on the InnoSAT swing and swing axes.

Keywords: Altitude regulation, Cuckoo search, Genetic algorithms, Nanosatellites

1. INTRODUCCIÓN

A finales de los años noventa, la Universidad Politécnica de California y la Universidad de Stanford desarrollaron el estándar "CubeSat" para promover el aprendizaje y la exploración espacial, este estándar permite el desarrollo de nanosatélites que tienen un tamaño, costo y tiempos de desarrollo reducidos en comparación con los satélites de tamaño regular, con un peso de no más de 1 kg [1]. Un ejemplo es el Satélite Innovador o InnoSat, un proyecto formado por Sains University Malaysia, Universiti Teknologi Malaysia y Universiti Malaysia Perlis, con el objetivo de tomar fotografías de diferentes regiones de Malasia desde órbita baja, almacenarlas y descargarlas una vez que entra en comunicación con la estación terrestre, sin embargo, también se pretende probar sistemas de altitud y sensores de paneles solares [2]. El InnoSat conceptualmente está compuesto por 3 CubeSats de dimensiones 10x10x10 cm³ en forma de torre con un peso máximo de 1 kg cada uno y un consumo total de energía de 15W, en la Figura 1 se puede ver una representación de un InnoSat. [3].



Figura 1. Esquema de un InnoSat.

Cabe señalar que el comportamiento dinámico de un InnoSAT se modela mediante las ecuaciones de Euler. Esto se muestra en las ecuaciones 1,2,3 [2] que corresponden los ángulos de Euler es decir a los rotacionales del satélite, ϕ corresponde al eje x , θ al eje Y , φ tiene su equivalencia en el eje z . Estos reciben el nombre de Roll, Pitch y Yaw y son representados por medio de la variable compleja s y la transformada de Laplace.

$$\varphi(s) = \frac{s^2 - 0.3023s + 0.8088}{s^4 + 1.1050s^2 + 0.1650} \quad (1)$$

$$\theta(s) = \frac{1}{s^2 - 7.1138 \times 10^{-3}} \quad (2)$$

$$\phi(s) = \frac{s^2 + 0.3015s + 0.240}{s^4 + 1.1050s^2 + 0.1650} \quad (3)$$

Los InnoSats son susceptibles de verse afectados por perturbaciones ambientales como la gravedad, el viento solar o la radiación, lo que provoca cambios bruscos en su posicionamiento, interfiriendo así en la misión a realizar, especialmente cuando se requiere un alto nivel de precisión, como en el caso de la captura fotográfica [4]. Para compensar el problema, los nanosatélites cuentan con un Sistema de Control de Actitud (ACS), que les permite regular su posición y orientación en órbita y compensar las perturbaciones derivadas del entorno.

Una forma de realizar ACS es a través de técnicas de control clásicas como el uso de reguladores tipo PID. Una alternativa pueden ser los reguladores inteligentes o algunas técnicas de control modernas, como es el caso de los reguladores cuadráticos

lineales (LQR), que se utilizan en una variedad de aplicaciones de control, incluidas la aeroespacial, la robótica, la ingeniería eléctrica y la mecánica. También se utilizan en sistemas de control de estabilidad, navegación y seguimiento.

Para entender un regulador LQR es necesario definir la dinámica de un sistema por medio de variables de estado, entendiendo el vector de estado \dot{x} definido como $\dot{x} = Ax + Bu$ con una salida que se puede describir como $y=Cx$ donde $x \in \mathbb{R}^n$ y $u \in \mathbb{R}^m$ a partir de esto se puede definir una función de costo J que se puede escribir como se muestra en la ecuación 4 de otra forma ,la ley de control $u(t)$ que minimiza la ecuación 4 puede describirse por medio de la ecuación 5 [6].

$$J = \int_0^{\infty} [x^T(t)Qx(t) + Ru^T(t)]dt = Cx \quad (4)$$

$$u(t) = -Kx(t) \quad (5)$$

Donde K es calculable por medio de la expresión mostrada en la ecuación 6, donde P es definida positiva y solución a la ecuación de Riccati definida como, $A^T P + AP - PBR^{-1}B^T P + Q = 0$ para $Q = Q^T \geq 0$, de manera adicional es real, simétrica y semidefinida positiva mientras que la matriz $R = R^T > 0$, is real, simétrica y positiva definida. La figura 2 muestra un esquema de regulación LQR donde el cálculo para K es realizado por medio de la ecuación 6 [6].

$$K = R^{-1}B^T P \quad (6)$$

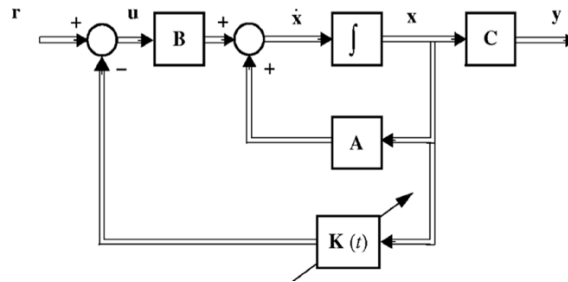


Figura 2. Esquema de variables de estado

2. SOPORTE TEÓRICO

Un regulador lineal cuadrado (LQR) está destinado a garantizar la estabilidad de un sistema de circuito cerrado mediante ciertas ganancias de retroalimentación. Este tipo de técnica de control pertenece a los métodos de control óptimos ya que se centran en la minimización de una función matemática llamada función de costo, definida como la suma de las desviaciones de los valores medidos con respecto a los deseados. Pero en el caso de InnoSat, se observa un error de estado estacionario considerable, que en varias posiciones teóricas puede mejorarse mediante técnicas de optimización como los algoritmos evolutivos.

En este trabajo, a partir de la información aportada en la sección anterior, **se propone un método de calculo de las ganancias K** , utilizando métodos heurísticos, como alternativa a la optimización de la ecuación de Riccati y de la ecuación 6. Con base en el estado del arte, este artículo propone comparar el rendimiento de un regulador de tipo LQR, ajustado por un algoritmo genético con codificación real, con respecto a una heurística recientemente desarrollada llamada Búsqueda Cuckoo, un método matemáticamente apoyado en la Teoría del Caos.

Como se muestra en la ecuación 6, el ajuste de las ganancias se puede realizar proponiendo las matrices y mediante Algoritmos Evolutivos (EAs), que se definen como una colección de métodos de resolución de problemas de optimización inspirados en los principios biológicos desarrollados por Charles Darwin y Gregory Mendel en el libro "El Origen de las Especies". Esto permitió a David Goldberg y John Holland proponer el primer Algoritmo Genético (AG) e Ingo Rechenberg las primeras estrategias evolutivas [7].

Los AE operan a partir de una colección de vectores $p, \vec{x} = \{x_1, x_2, x_3, \dots, x_n\}$ que constituyen las posibles soluciones a un problema de optimización que es modelable por medio de una función matemática $f_j(\vec{x})$ donde x_n representa la n -ésima variable de un j -ésimo problema, tal que $\vec{x} \in \mathbb{R} \parallel \vec{x} \in \mathbb{Z}$ para problemas de optimización numérica o combinatoria respectivamente. Algunos elementos de p se caracterizan por tener el valor mas alto de $f_j(\vec{x})$ están sujetos a varios procesos matemáticos de manera iterativa. En un GA, una de las operaciones características es el cruzamiento, que se muestra en las expresiones 7 y 8, cabe destacar que la p colección se denomina cromosomas y cada uno de sus componentes se denominan genes y representan una variable del problema de optimización a resolver [8].

$$C_n^{H1} = \beta * C_n^1 + (1 - \beta) * C_n^2 \quad (7)$$

$$C_n^{H2} = \beta * C_n^2 + (1 - \beta) * C_n^1 \quad (8)$$

Donde $\beta = rand[0,1]$, C_n^1 y C_n^2 . Son cromosomas a recombinar, C_n^{H1} y C_n^{H2} . Son los cromosomas recombinados. Otro de los operadores de recombinación GA es la mutación, que es un proceso que tiene como objetivo cambiar un gen de un cromosoma, elegido al azar, a través de una distribución uniforme dentro de un intervalo definido en los límites superior e inferior de dicha variable, este operador se denomina mutación aleatoria. El algoritmo 1 muestra el diagrama de flujo de la AG que se implementó en esta propuesta.

Algoritmo 1 Algoritmo Genético	
1	Inicializar población
2	Calcular función de aptitud
3	While $i <$ maximo número de iteraciones do
4	Seleccionar cromosomas padres
5	Cruzar nuevos candidatos
6	Mutar individuo para la proxima generación
	Remplazar
7	end
8	end

Cabe destacar que la propuesta desarrollada por Holland posibilitó, posteriormente, desarrollar una variedad de métodos bioinspirados para resolver problemas de optimización, uno de ellos es el Algoritmo de Búsqueda de Cuckoo (CSA), que teóricamente se apoya en el comportamiento del Cuckoo, un ave depredadora endémica de Europa y el norte de África que tiene hábitos migratorios hacia el subcontinente asiático. El Cuckoo, en su ciclo reproductivo, tiene un comportamiento parasitario

que consiste en poner en nidos de otras especies sus huevos, que son incubados al camuflarse. El CSA fue propuesto en 2019 por She Yang de la Universidad de Cambridge y Suash Deb de la Universidad CV Raman en India. Básicamente, este algoritmo realiza una exploración y actualización de los nidos, que son posibles soluciones, por medio de un generador de números aleatorios con distribución caótica, que se lleva a cabo por medio de una función matemática llamada vuelo de Levy en honor al matemático francés Pierre Lévy, quien describió los movimientos de las aves por medio de un comportamiento de tipo fractal. Esto se muestra en la ecuación 8 [9].

$$\sigma_u = \left[\frac{T(1 + \beta) * \sin\left(\frac{\pi * \beta}{2}\right)}{T\left(\frac{1 + \beta}{2}\right) * \beta * 2^{(\beta+1)/2}} \right]^{\frac{1}{\beta}} \quad (8)$$

Donde $\beta = [0.25,3]$, es el tamaño del paso que determina el espacio de búsqueda, T es la función Gamma, la dimensión del generador de números aleatorios está determinada por $S = \frac{u}{|v|^{1/\beta}} \forall u \sim N(0, \sigma_u^2), v \sim N(0,1)$. Con base en lo anterior, es posible actualizar los nidos mediante la expresión que se muestra en la ecuación 9 [9].

$$nest_i^{(t+1)} = nest_i^{(t)} + \alpha * S(\max(nest^t) - \min(nest^t)) * r \quad (9)$$

Donde, $nest_i^{(t)}$ es el nido actual, mientras $nest_i^{(t+1)}$ Es el nido actualizado, además r es un número aleatorio entre 0 y 1, α es un parámetro entre 2 y 4. El CSA selecciona anidamientos aleatoriamente, los actualiza y elige nuevos nidos que se reemplazarán si su valor cuando se evalúa en la función de destino es menor que el de los nidos originales. El algoritmo 3 muestra el diagrama de flujo del CSA que se implementó en esta propuesta.

Algoritmo2 Algoritmo de Búsqueda Cuckoo	
1	Inicializar población
2	Calcular función de aptitud
3	While $i <$ maximo número de iteraciones do
4	Consigue un cuco al azar en los vuelos de Levy
5	Evaluar aptitud de los nuevos candidatos $f(\vec{x}_i)$
6	Elija un nido entre n (digamos, j) al azar
7	If $F_i > F_j$ then
8	Reemplazar por la nueva soluciónj
9	End if
10	Una fracción (Pa) de los peores nidos se abandonan y se construyen otros nuevos
11	Mantener las mejores soluciones
12	Clasificar las soluciones y encuentre las mejores
13	End While

3. METODOLOGÍA

En este proceso de investigación se propone el diseño de LQR's a partir de algoritmos heurísticos para los ejes Yaw y Roll, las funciones objetivo a optimizar en este proceso de investigación

son las ecuaciones 10 y 11 para φ y ϕ , respectivamente. Donde ITAE es la integral del error absoluto multiplicado por el tiempo, IAE es la integral del error absoluto y ISE es la integral del error al cuadrado.

$$f_{obj\varphi} = \frac{1}{ITAE_{\varphi} + IAE_{\varphi} + ISE_{\varphi}} \quad (10)$$

$$f_{obj\phi} = \frac{1}{ITAE_{\phi} + IAE_{\phi} + ISE_{\phi}} \quad (11)$$

De acuerdo con lo mostrado en el apartado correspondiente a la fundamentación teórica, es necesario representar las funciones de transferencia expresadas en 1,2,3 como un sistema de matrices de estado, esto se muestra en la tabla 1. Dada la representación que se muestra en la Tabla 1, es necesario configurar las posibles soluciones, $\vec{x} = \{Q_{11}, Q_{22}, Q_{33}, Q_{44}, R_{11}\}$ como se muestra en la Tabla 2 para el diseño de los reguladores mostrados en ejes φ y ϕ , que son objeto de estudio en la propuesta que se documenta en este trabajo.

Tabla 1 Variables de estado del sistema

matriz Angulos	A	B	C	D
φ	$\begin{bmatrix} 0 & -1.105 & 0 & -0.1650 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$	$[0 \ 1 \ -0.3023 \ 0.8088]$	$[0]$
θ	$\begin{bmatrix} 0 & 0.0071 \\ 1 & 0 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0 \end{bmatrix}$	$[0 \ 1]$	$[0]$
ϕ	$\begin{bmatrix} 0 & -1.105 & 0 & -0.1650 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$	$[0 \ 1 \ 0.3051 \ 0.2040]$	$[0]$

Table 2. Variables de estado del controlador

Matriz Angulos	Q	R
φ	$\begin{bmatrix} Q_{11} & 0 & 0 & 0 \\ 0 & Q_{22} & 0 & 0 \\ 0 & 0 & Q_{33} & 0 \\ 0 & 0 & 0 & Q_{44} \end{bmatrix}$	$[r_{11}]$
ϕ	$\begin{bmatrix} Q_{11} & 0 & 0 & 0 \\ 0 & Q_{22} & 0 & 0 \\ 0 & 0 & Q_{33} & 0 \\ 0 & 0 & 0 & Q_{44} \end{bmatrix}$	$[r_{11}]$

En la tabla 3 se muestran las características operacionales de los algoritmos utilizados en esta propuesta, que fueron las mismas para los ángulos φ y ϕ .

Tabla 3. Características de los algoritmos propuestos

	GA y CSA
Tamaño de la población (posibles soluciones)	100
Rango de población	$Q_{11} = rand[0,10]$ $Q_{22} = rand[0,10]$ $Q_{33} = rand[0,10]$ $Q_{44} = rand[0,10]$ $R_{11} = rand[0,10]$

4. RESULTADOS

Para el proceso de investigación documentado en esta propuesta, se realizaron comparaciones entre los ejes ψ y ϕ dado que, de acuerdo con la literatura especializada, presentan una mayor dificultad para el diseño de reguladores en comparación con los ejes del eje θ , que se caracteriza por ser una función de segundo orden. Las Tablas 4 y 5 muestran los siguientes criterios de desempeño: tiempo de establecimiento t_s , sobre impulso M_p y error en estado estable E_{SS} , además de los errores descritos en la función objetivo para los ángulos ψ y ϕ respectivamente. Los mejores valores obtenidos por los algoritmos están resaltados en rojo.

Tabla 4. Criterios de rendimiento de ángulo ψ

Criterio de desempeño	Ajuste con GA	Ajuste con CSA
t_s	10	10
M_p	1.0422	1.0422
ITAE	0.0169	0.0186
IAE	1.6888	1.8557
ISE	0.9888	0.9748
E_{SS}	0.0089	6.2960e-04

Tabla 5. Criterios de rendimiento del ángulo ϕ

Criterio de desempeño	Ajuste con Ga	Ajuste con CSA
t_s	20	16
M_p	1.2720	1.124
ITAE	0.0333	0.0343
IAE	3.3342	3.4324
ISE	2.0249	2.2465
E_{SS}	-3.51e-04	-2.6530e-04

Las matrices Q y R para los ejes ψ y ϕ obtenidas por el GA utilizado en esta propuesta se muestran en la tabla 6.

Tabla 6. Matrices Q y R obtenidas con un GA

	Q	R
ψ	$\begin{bmatrix} 0.0097 & 0 & 0 & 0 \\ 0 & 1.6210 & 0 & 0 \\ 0 & 0 & 0.0018 & 0 \\ 0 & 0 & 0 & 7.2515 \end{bmatrix}$	[11.5666]
ϕ	$\begin{bmatrix} 21.9797 & 0 & 0 & 0 \\ 0 & 0.0037 & 0 & 0 \\ 0 & 0 & 0.0035 & 0 \\ 0 & 0 & 0 & 0.3889 \end{bmatrix}$	[27.0171]

Las matrices Q y R para los ejes ψ y ϕ obtenidas por el CSA utilizado en esta propuesta se muestran en la tabla 6.

Tabla 7. Matrices Q y R obtenidas con un eje CSA

	Q	R
ψ	$\begin{bmatrix} 0.0494 & 0 & 0 & 0 \\ 0 & 0.0221 & 0 & 0 \\ 0 & 0 & 0.0107 & 0 \\ 0 & 0 & 0 & 0.5945 \end{bmatrix}$	[0.9483]
ϕ	$\begin{bmatrix} 3.9271 & 0 & 0 & 0 \\ 0 & 0.0003 & 0 & 0 \\ 0 & 0 & 0.0098 & 0 \\ 0 & 0 & 0 & 0.0422 \end{bmatrix}$	[4.5016]

Las Figuras 3 y 4 muestran las respuestas al escalón unitario para los ejes ψ y ϕ respectivamente. Por otro lado, la convergencia de los algoritmos GA y CSA, es decir, sus respuestas, se muestra en las Figuras 5 y 6 para los ejes ψ y ϕ

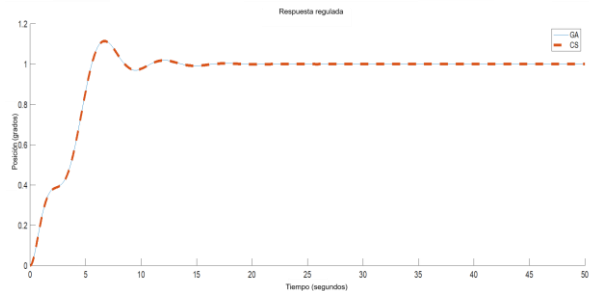


Figura 3. Respuesta regulada por LQR al paso unitario del ψ eje

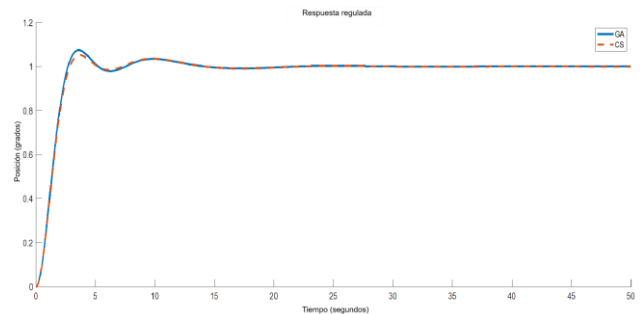


Figura 4. Respuesta regulada por LQR al paso unitario del eje ϕ

3. Proponer una representación de la rotación del InnoSAT como un modelo MIMO, de entrada múltiple y salida múltiple

Actualmente se están realizando esfuerzos en México para insertar al país en la exploración espacial, esfuerzos realizados por universidades públicas en coordinación con la agencia espacial mexicana, como el documentado en esta propuesta, están orientados a desarrollar tecnología propia.

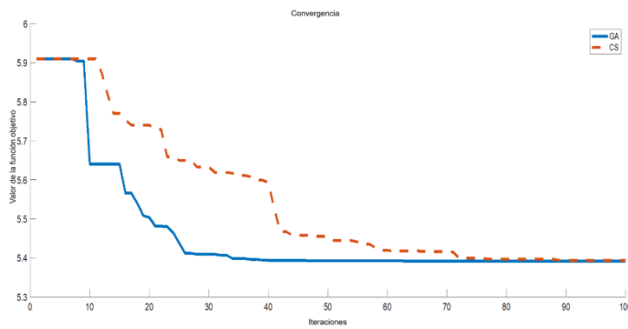


Figura 5. Convergencia de ejes ϕ

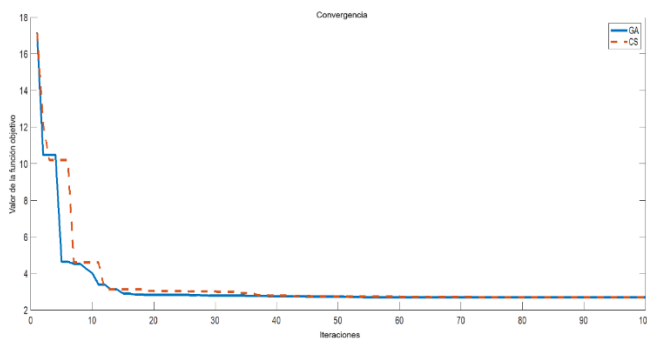


Figura 6. Convergencia de ejes ϕ

5. CONCLUSIONES

Como se puede observar en las cifras mostradas en los apartados anteriores de este proceso de investigación, el comportamiento de los controladores con AG y CSA es similar, pero se puede observar que el porcentaje de error en el estado estacionario presenta una amplia mejora con respecto al ajuste realizado mediante un AG. Por ello, como trabajo futuro sería necesario hibridar ambos métodos de resolución de un problema de optimización como el planteado en esta propuesta. A partir de los resultados mostrados, es posible establecer futuras líneas de actuación como las que se describen a continuación:

1. Realizar un análisis estadístico inferencial, que determinará si existen diferencias estadísticamente significativas
2. Prueba de nuevos métodos heurísticos

REFERENCIAS

- [1] S.M. Sharun, M.Y.Mashor, M.N. Norhayati, Sazali, Yaacob, Muhyi Yaakob (2011). ADAPTIVE NEURO-CONTROLLER BASED ON HMLP NETWORK FOR INNOSAT ATTITUDE CONTROL. International Conference on Electrical, Control and Computer Engineering. pp. 355-360, doi: 10.1109/INECCE.2011.5953906
- [2] Sharun, S. M., Mashor, M. Y., Mohd Nazid, N., Yaacob, S., & Wan Jaafar, W. N. (2011). INNOSAT ATTITUDE CONTROL SYSTEM BASED ON ADAPTIVE NEURO-CONTROLLER. Journal of Information and Communication Technology, 10, 45–65. Retrieved from <https://e-journal.uum.edu.my/index.php/jict/article/view/8108>
- [3] Fadly, M., & Sidek, O. (2010). Development of Attitude Determination for Student Pico-Satellite INNOSAT. Regional Conference on Mechanical and Aerospace Technology. Bali, Indonesia.
- [4] Biblal, M., Vijayan, R., Schilling, K.(2019). SDRE CONTROL WITH NONLINEAR J2 PERTURBATIONS FOR NANOSATELLITE FORMATION FLYING, IFAC-PapersOnLine, Volume 52, Issue 12, 2, Pages 448-453, <https://doi.org/10.1016/j.ifacol.2019.11.284>
- [5] H. H. A., M. Y. Mashor y M. C. Mahdi, “Performance of Manual and Auto-Tuning PID Controller for Unstable Plant - Nano Satellite Attitude Control System”, en 2018 6th Int. Conf. Cyber IT Service Manage. (CITSM), Parapat, Indonesia, 7–9 de agosto de 2018. IEEE, 2018. Accedido el 22 de septiembre de 2024. [En línea]. Disponible: <https://doi.org/10.1109/citsm.2018.8674375>
- [6] J. A. B. Quispe, R. J. H. Kemper y S. R. P. Gardini, “Linear Quadratic Regulator (LQR) Control for the Active Suspension System of a Four-Wheeled Agricultural Robot”, en 2023 IEEE XXX Int. Conf. Electron., Elect. Eng. Comput. (INTERCON), Lima, Peru, 2–4 de noviembre de 2023. IEEE, 2023. Accedido el 23 de septiembre de 2024. [En línea]. Disponible: <https://doi.org/10.1109/intercon59652.2023.10326049>
- [7] I. Rechenberg, “Evolution, robotics and the somersaulting spider”, en GECCO '19: Genetic Evol. Computation Conf., Prague Czech Republic. New York, NY, USA: ACM, 2019. Accedido el 23 de septiembre de 2024. [En línea]. Disponible: <https://doi.org/10.1145/3321707.3326936>
- [8] G. Singh y N. Gupta, “A Study of Crossover Operators in Genetic Algorithms”, en Springer Tracts in Nature-Inspired Computing. Singapore: Springer Singap., 2021, pp. 17–32. Accedido el 23 de septiembre de 2024. [En línea]. Disponible: https://doi.org/10.1007/978-981-16-3128-3_2
- [9] M. Shehab, A. T. Khader y M. A. Al-Betar, “A survey on applications and variants of the cuckoo search algorithm”, Appl. Soft Comput., vol. 61, pp. 1041–1059, diciembre de 2017. Accedido el 23 de septiembre de 2024. [En línea]. Disponible: <https://doi.org/10.1016/j.asoc.2017.02.034>